Texas Instruments Technology for Innovators™

TMS320DM644x™ Processors–Video Benchmarks

Dedicated video processing sub-system incorporates:

- Back end – Integrated OSD, four video DACs, 24-bit digital RGB output
- Front end – Resizer, image processing engine, 16-bit digital input *(DM6446 processor only)*

TMS320DM644x Processor Video Capabilities

	TMS320DM6446	TMS320DM6443
STANDALONE CODECS		
MPEG-2 MP ML decode	720p (30 fps)	720p (30 fps)
MPEG-2 MP ML encode	D1[+]	n/a
MPEG-4 SP decode	720p (30 fps)	720p (30 fps)
MPEG-4 SP encode	D1[+]	n/a
VC1/WMV 9 decode	720p (30 fps)	720p (30 fps)
VC1/WMV 9 encode	D1[+]	n/a
H.264 (Baseline) decode	D1[+]	D1[+]
H.264 (Baseline) encode	D1[+]	n/a
H.264 (Main Profile) decode	D1[+]	D1[+]

[+] Denotes available processor headroom for analytics and/or other features
Encode is only available on the DM6446 processor.
Note: Performance will vary depending on efficiency of code and data stream used.
Resolution information: D1 (720×480) / 720p (1280×720)

All performance data is for 30-fps YUV 4:2:0 unless otherwise noted.
SP = Simple Profile / MP= Main Profile

The TMS320DM644x devices, available today, are based on the TMS320C64x+™ DSP core. TMS320C64x+ DSP core benchmarks include:

Filters

Benchmark	Description	Formula
Complex FIR filter	Computes a complex FIR filter (direct-form) with nh coefficients and nr output samples. Nh and nr must be a multiple of 4.	nh * nr / 2 + 16 For nh = 32 and nr = 100: cycles = 1616
FIR filter	Computes a real FIR filter (direct-form) with nh coefficients and nr output samples. Nh and nr must be a multiple of 8.	T ≥ 32: nh * nr / 8 + 22 Other: 32 * nr / 8 + 22 For nh = 32 and nr = 100: cycles = 422
IIR biquad	Performs single biquad IIR filter for nx samples.	nx*4 + 25 For nx = 16: cycles = 89
Autocorrelation	Performs nr autocorrelations, each of length nx, producing nr output results.	nx < nx ≥ 40: 20 + (2 * nr) + (nx * nr / 8) For nr = 160, nx = 40: cycles = 1140

(over)

FFTs

Benchmark	Description	Formula
Complex, forward FFT (radix 4) with digit reversal	Computes a complex forward radix-4 nx-point FFT. Input data, output data and coefficients are 16-bit.	0.75*nx*log4(nx) + 38 For nx = 1024: cycles = 3878
Extended-precision, mixed-radix 16×32 FFT with rounding, digit reversal	Computes an extended-precision complex forward mixed-radix nx-point FFT with rounding and digit reversal. Input data and output data are 32-bit, coefficients are 16-bit.	[10.25*nx/8+10]*ceil[log4(nx) – 1] + 6*nx/4 + 81 For nx = 124: cycles = 6905
Extended-precision, mixed-radix 32×32 FFT with rounding, digit reversal	Computes an extended-precision complex-forward mixed-radix nx-point FFT with rounding and digit reversal. Input data, output data and coefficients are 32-bit.	[12*nx/8+12]*ceil[log4(nx) – 1] + 6*nx/4 + 47 For nx = 1024: cycles = 7775

Vector

Benchmark	Description	Formula
Vector dot product	Computes dot-product of two vectors of size nx elements.	nx/4 + 14 For nx = 100: cycles = 39
Vector sum	Computes and nx-element vector sum of two vectors. The result is stored in a third vector.	3*(nx/8) + 10 For nx = 256: cycles = 106

Search

Benchmark	Description	Formula
Maximum value of a vector	Finds the element with maximum value in a vector of size nx.	nx/8 + 13 For nx = 256: cycles = 45
Index of the maximum element of a vector	Finds the index of the element with the maximum value in a vector of size nx.	nx/4 + 20 For nx = 100: cycles = 45

Image/Video Compression/Decompression

Benchmark	Description	Formula
8×8 block forward discrete cosine transfrom (FDCT)	Computes a series of num_fdcts 8×8 forward discrete cosine transforms (FDCT)	num_fdcts * 52 + 56 For num_fdcts = 6: cycles = 368
8×8 block inverse discrete cosine transform (IDCT)	Computes a series of num_idcts IEEE 1180 - 1990 compliant 8×8 inverse discrete cosine transforms (IDCT)	num_idcts * 72 + 63 For num_idcts = 6: cycles = 495

Telecom

Benchmark	Description	Formula
convenc3	Implements rate=1/3, R=9 convolutional encoding.	cycles = 14 + 3*ceil((nbits+8)/32) For nbits = 512: cycles = 65
crc32	Compute 32-bit cyclic redundancy check (CRC) of the input data.	14 + N/2, N = num of Bytes For N = 128: cycles = 78

Important Notice: The products and services of Texas Instruments Incorporated and its subsidiaries described herein are sold subject to TI's standard terms and conditions of sale. Customers are advised to obtain the most current and complete information about TI products and services before placing orders. TI assumes no liability for applications assistance, customer's applications or product designs, software performance, or infringement of patents. The publication of information regarding any other company's products or services does not constitute TI's approval, warranty or endorsement thereof.

Technology for Innovators, the black/red banner, TMS320DM644x and TMS320C64x+ are trademarks of Texas Instruments. All other trademarks are the property of their respective owners.

© 2006 Texas Instruments Incorporated
Printed in U.S.A. by C2 Media, Houston, TX